



PATENT ABSTRACTS OF JAPAN

(11) Publication number: **05197551 A**(43) Date of publication of application: **06 . 08 . 93**

(51) Int. Cl.

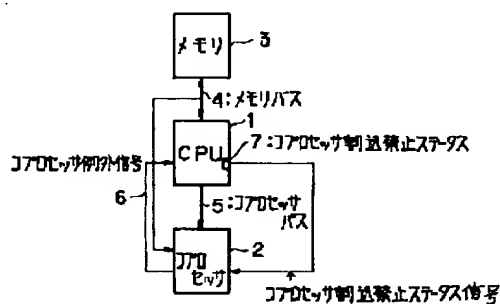
G06F 9/38
G06F 9/46
(21) Application number: **04008907**(22) Date of filing: **22 . 01 . 92**(71) Applicant: **FUJITSU LTD**
(72) Inventor: **MARUYAMA TAKUMI**
NODA TAKAHITO
KAMISAKA YUJI
NONOMURA KAZUYASU
WATABE TORU
TAKENO TAKUMI
KATO SHINYA
POONSHIYAI
CHIYONSUWANNAPAJSAAN
(54) **COPROCESSOR CONTROL SYSTEM**

COPYRIGHT: (C)1993,JPO&Japio

(57) Abstract:

PURPOSE: To minimize a decrease in the performance of a CPU by providing a coprocessor interruption-inhibiting status without increasing the widths of buses between a CPU and a coprocessor and operating it.

CONSTITUTION: This system is equipped with the coprocessor 2 and the coprocessor interruption-inhibiting status 7 which inhibits and allows the transmission of a coprocessor exception signal in the case of that exception is generated. The CPU 1 inputs and decodes instructions in order and sends only a coprocessor instruction to the coprocessor 2 on condition that the coprocessor interruption-inhibiting status 7 is ON once deciding the coprocessor instruction, thereby inhibiting the address of the coprocessor instruction from being sent out. When the coprocessor interruption-inhibiting status 7 is OFF, on the other hand, the coprocessor instruction and its address are sent out to the coprocessor 2.



(19)日本国特許庁 (J P)

(12)特 許 公 報 (B 2)

(11)特許番号

第 2 9 0 8 0 9 6 号

(45)発行日 平成 1 1 年 (1 9 9 9) 6 月 2 1 日

(24)登録日 平成 1 1 年 (1 9 9 9) 4 月 2 日

(51)Int.Cl. ⁶	識別記号	庁内整理番号	F I	技術表示箇所
G06F 9/38	370		G06F 9/38	370 C
	380			380 B

請求項の数 3 (全 8 頁)

(21)出願番号	特願平 4 - 8 9 0 7
(22)出願日	平成 4 年 (1 9 9 2) 1 月 2 2 日
(65)公開番号	特開平 5 - 1 9 7 5 5 1
(43)公開日	平成 5 年 (1 9 9 3) 8 月 6 日

(73)特許権者	0 0 0 0 0 5 2 2 3 富士通株式会社 神奈川県川崎市中原区上小田中 4 丁目 1 番 1 号
(72)発明者	丸山 拓巳 神奈川県川崎市中原区上小田中 1 0 1 5 番地 富士通株式会社内
(72)発明者	野田 敬人 神奈川県川崎市中原区上小田中 1 0 1 5 番地 富士通株式会社内
(72)発明者	神阪 裕士 神奈川県川崎市中原区上小田中 1 0 1 5 番地 富士通株式会社内
(74)代理人	弁理士 岡田 守弘
審査官	中野 裕二

最終頁に続く

(54)【発明の名称】 コプロセッサ制御方式

1

(57)【特許請求の範囲】

【請求項 1】 CPU からコプロセッサを制御するコプロセッサ制御方式において、
CPU (1) がコプロセッサ命令の実行を依頼するコプロセッサ (2) と、
コプロセッサ (2) がコプロセッサ命令実行によって例外の発生したときにコプロセッサ例外信号の送出を禁止／許可するコプロセッサ割込禁止ステータス (7) とを備え、
CPU (1) が命令を順次取り込んでデコードしてコプロセッサ命令と判明したときに上記コプロセッサ割込禁止ステータス (7) が ON のときにコプロセッサ命令のみをコプロセッサ (2) に送出し、当該コプロセッサ命令のアドレスの送出を抑制し、一方、上記コプロセッサ割込禁止ステータス (7) が OFF のときにコプロセッ

2

サ命令および当該コプロセッサ命令のアドレスをコプロセッサ (2) に送出するように構成したことを特徴とするコプロセッサ制御方式。

【請求項 2】 上記 CPU (1) からコプロセッサ命令の送出を受けてこれを取り込んだコプロセッサ (2) が実行して例外発生したとき、上記コプロセッサ割込禁止ステータス (7) が ON のときに当該例外を CPU (1) へ通知することを抑制し、一方、上記コプロセッサ割込禁止ステータス (7) が OFF のときに当該例外を CPU (1) へ通知し、この通知を受けた CPU (1) がコプロセッサ (2) の保持するデータをもとに所定の割り込み処理を行うように構成したことを特徴とする請求項第 1 項記載のコプロセッサ制御方式。

【請求項 3】 上記コプロセッサ (2) が実行して例外発生したとき、上記コプロセッサ割込禁止ステータス

(7) が ON のときに当該例外を CPU (1) へ通知することを抑止すると共に、例外処理した結果を実行結果として格納するように構成したことを特徴とする請求項第 2 項記載のコプロセッサ制御方式。

【発明の詳細な説明】

【0001】

【産業上の利用分野】 本発明は、CPU からコプロセッサを制御するコプロセッサ制御方式に関するものである。

【0002】 近年のコンピュータシステムの高速化の要求に伴い、CPU の他にコプロセッサを設けるシステムが一般的になっている。このシステムは、CPU がフェッチした命令のうち、特定の命令（以下コプロセッサ命令という）について、コプロセッサに渡してコプロセッサに実行させ、その結果を受け取るものである。この際、処理の高速化が要求されている。

【0003】

【従来の技術】 従来、CPU の他にコプロセッサを設けたシステムとして、図 4 の (a) に示すような構成を持つシステムがある。以下説明する。

【0004】 図 4 の (a) において、CPU 31 は、処理装置であって、メモリ 33 から読み込んだ命令をデコードして実行するものである。コプロセッサ 32 は、コプロセッサ命令を実行するものである。

【0005】 メモリ 33 は、命令やデータを保持するものである。メモリバス 34 は、メモリ 33 から CPU 31 へ命令転送したり、メモリ 33 と CPU 31 やコプロセッサ 32 の間のデータ転送したりするものである。

【0006】 コプロセッサバス 35 は、CPU 31 とコプロセッサ 32 の間の命令コード／アドレス転送するものである。コプロセッサ例外信号 36 は、コプロセッサ 32 がコプロセッサ命令の実行中の例外事象（例えば浮動小数点演算コプロセッサのときはオーバーフローやアンダーフローという例外事象）の発生時に CPU 31 に通知するものである。

【0007】 次に、図 4 の (b) のタイムチャートに従い、CPU 31 の命令実行は以下のシーケンスで行う。

(1) 必要な命令をメモリ 33 からメモリバス 34 を通してフェッチする（ステージ F）。

【0008】 (2) 命令をデコードする（ステージ D）。

(3) 命令を実行する（ステージ E）。

(3-1) デコード結果がコプロセッサ命令でなければ（以下一般命令という）、CPU 31 がこの一般命令を実行する。

【0009】 (3-2) デコード結果がコプロセッサ命令であれば、コプロセッサ命令をコプロセッサバス 35 を通してコプロセッサ 32 に送出する。

【0010】 次のサイクルで割り込み処理に必要な資源であるこのコプロセッサ命令のアドレスを、予めコ

ロセッサバス 35 を通してコプロセッサ 32 に送出しておく。

【0011】 (4) 結果を格納する（ステージ W）。一方、コプロセッサ 32 のコプロセッサ命令実行は以下のシーケンスで行う。

(11) 命令を実行する（ステージ CE）。

【0012】 (12) 結果を格納する（ステージ CW）。

ここで、ステージ CW 終了後にコプロセッサ例外（例えば浮動小数点演算コプロセッサのときはオーバーフローやアンダーフローという例外）が検出された場合、コプロセッサ 32 はコプロセッサ例外信号 36 を通して CPU 31 に例外発生を割り込みで通知する。CPU 31 は、コプロセッサ割り込み許可状態ならば、当該通知により割り込みを発生する。そして、割り込み処理ルーチンで、コプロセッサ 32 に保持されている命令コードおよび命令のアドレスを用い、所定の割り込み処理を行う。

【0013】 コプロセッサ割り込み禁止状態ならば、当該通知を無視する。

20 【0014】

【発明が解決しようとする課題】 上述したように、従来は、CPU 31 からコプロセッサ 32 に単一バスであるコプロセッサバス 35 を通してコプロセッサ命令および当該命令のアドレスを常に順次送出しており、CPU 31 がこれら両者のためのサイクル必要となってしまう、性能を低下させてしまうという問題があった。

【0015】 本発明は、これらの問題を解決するため、CPU とコプロセッサ間のバス幅を増加させることなく、コプロセッサ割込禁止ステータスを設けてこれを操作し、CPU の性能低下を必要最小限に抑えることを目的としている。

【0016】

【課題を解決するための手段】 図 1 は、本発明の原理構成図を示す。図 1 において、CPU 1 は、命令を読み込んでデコードし、コプロセッサ命令のときにコプロセッサ 2 に実行を依頼し、それ以外の一般命令のときに実行などするものである。

【0017】 コプロセッサ 2 は、CPU 1 から依頼を受けたコプロセッサ命令を実行などするプロセッサである。メモリ 3 は、命令やデータを保持するものである。

【0018】 メモリバス 4 は、メモリ 3 と CPU 1 やコプロセッサ 2 の間で命令やデータの転送を行うものである。コプロセッサバス 5 は、CPU 1 からコプロセッサ 5 にコプロセッサ命令などを送出するバスである。

【0019】 コプロセッサ例外信号 6 は、コプロセッサ 2 がコプロセッサ命令を実行して例外が発生したときにこれを CPU 1 に通知するものである。コプロセッサ割込禁止ステータス 7 は、コプロセッサ 2 がコプロセッサ命令実行によって例外の発生したときにコプロセッサ例外信号を CPU 1 に対しての送出を禁止／許可するステ

ータスである。

【0020】

【作用】本発明は、図1に示すように、CPU1がメモリ3から命令をメモリバス4を通して順次取り込んでデコードしてコプロセッサ命令と判明したとき、コプロセッサ割込禁止ステータス7がONのときにコプロセッサ命令のみをコプロセッサバス5を通してコプロセッサ2に送出し、当該コプロセッサ命令のアドレスの送出を抑制し、一方、コプロセッサ割込禁止ステータス7がOFFのときにコプロセッサ命令および当該コプロセッサ命令のアドレスをコプロセッサ2に送出するようにしている。

【0021】また、CPU1からコプロセッサ命令の送出を受けてこれを取り込んだコプロセッサ2が実行して例外発生したとき、コプロセッサ割込禁止ステータス7がONのときに当該例外をCPU1へ通知することを抑制すると共に例外処理を行ってその結果を実行結果として格納し、一方、コプロセッサ割込禁止ステータス7がOFFのときに当該例外（コプロセッサ例外信号6）をCPU1へ通知し、この通知を受けたCPU1がコプロセッサ2の保持するコプロセッサ命令およびそのアドレスをもとに所定の割り込み処理を行うようにしている。

【0022】従って、CPU1とコプロセッサ2の間のバス幅を増加させることなく、コプロセッサ割込禁止ステータス7を設けてこれを利用して無用な命令アドレスの送出を無くし、CPU1の性能低下を防止することが可能となる。

【0023】

【実施例】次に、図2および図3を用いて本発明の実施例の構成および動作を順次詳細に説明する。

【0024】図2は、本発明の1実施例構成図を示す。図2において、CPU1は、メモリ3からメモリバス4を通して命令を読み込んでデコードし、各種処理を行うものである。ここでは、デコードしてコプロセッサ命令と判明したとき、コプロセッサ割込禁止ステータス7がONのときにコプロセッサ命令のみをコプロセッサバス5を通して、ここでは浮動小数点コプロセッサ21に送出して実行依頼し、一方、コプロセッサ割込禁止ステータス7がOFFのときにコプロセッサ命令および当該命令のアドレスを併せてコプロセッサバス5を通して浮動小数点コプロセッサ21に送出して実行依頼する。そして、浮動小数点コプロセッサ21がコプロセッサ命令を実行して例外（オーバーフロー、アンダーフロー）が発生したとき、コプロセッサ割込禁止ステータス7がONのとき、コプロセッサ例外信号6を抑制すると共に例外処理を行った結果を実行結果としてメモリ3に格納する。一方、コプロセッサ割込禁止ステータス7がOFFのとき、コプロセッサ例外信号6をCPU1に通知し、割り込みを受け付けたCPU1が浮動小数点コプロセッサ21が保持するコプロセッサ命令および命令のアドレスを参照し、所

定の割り込み処理を行う。

【0025】浮動小数点コプロセッサ21は、CPU1からコプロセッサバス5を通してコプロセッサ命令を受け取り、このコプロセッサ命令を実行するものである。実行した結果、例外（オーバーフロー、アンダーフロー）が発生した場合には、コプロセッサ割込禁止ステータス7がONのときは、コプロセッサ例外信号6をCPU1に送出することを抑制すると共に例外処理として最大値あるいは零を実行結果としてメモリ3に格納する。一方、コプロセッサ割込禁止ステータス7がOFFのときは、コプロセッサ例外信号6をCPU1に送出する。

【0026】メモリ3は、命令やデータを保持するものである。メモリバス4は、メモリ3から読み出した命令をCPU1に転送したり、CPU1から送出されたデータをメモリ3に転送したりなどするものである。

【0027】コプロセッサバス5は、CPU1から送出されたコプロセッサ命令や命令アドレスなどを浮動小数点コプロセッサ21に転送したりするものである。コプロセッサ例外信号6は、コプロセッサ割込禁止ステータス7がOFFのときに浮動小数点コプロセッサ21が例外発生時にCPU1に送出する信号である。

【0028】コプロセッサ割込禁止ステータス7は、浮動小数点コプロセッサ21からの例外割込を禁止するステータスであって、CPU1が取り込んでデコードした命令がコプロセッサ命令であって、これがONのときにコプロセッサ命令のみを浮動小数点コプロセッサ21に送出し、命令アドレスの送出を抑制する。一方、これがOFFのときにコプロセッサ命令および当該命令のアドレスの両者を浮動小数点コプロセッサ21に送出する。そして、浮動小数点コプロセッサ21が依頼を受けたコプロセッサ命令を実行し、例外（オーバーフロー、アンダーフロー）が発生したときに、コプロセッサ割込禁止ステータス7がONのとき、コプロセッサ例外信号6をCPU1へ送出することを抑制すると共に例外処理を行う。一方、コプロセッサ割込禁止ステータス7がOFFのとき、コプロセッサ例外信号6をCPU1へ送出し、CPU1が割込処理を行う。

【0029】レジスタ11は、コプロセッサ割込禁止ステータス7を設定するものである。このレジスタ11には、応用プログラムなどが予めビットをON/OFFし、コプロセッサ割込禁止ステータス7をON/OFFする。

【0030】次に、図3のタイムチャートを用いて図2の構成の動作を詳細に説明する。図3の(a)は、コプロセッサ割込禁止状態（コプロセッサ割込禁止ステータス7がONの場合）のタイムチャートである。ここで、斜線の部分が本実施例のCI（コプロセッサ命令）である。

【0031】図3の(a)において、ステージF：CP

U1がメモリ3からメモリバス4を通して斜線を引いたコプロセッサ命令C I ②をフェッチする(読み込む)。ステージD:読み込んだコプロセッサ命令C I ②をデコードし、ここではコプロセッサ命令と判明すると共に、レジスタ11のコプロセッサ割込ステータス7のONを読み取って認識する。

【0032】ステージE:命令を実行する。ここでは、ステージDでコプロセッサ命令と判明し、かつコプロセッサ割込ステータス7がONと判明したので、このコプロセッサ命令C I ②のみを、コプロセッサバス5を通して浮動小数点コプロセッサ21に送出する。この際、コプロセッサ割込ステータス7がONであったので、当該コプロセッサ命令C I ②のアドレスの送出を抑止する。

【0033】ステージW:一般命令では結果を格納するが、ここではしない。結果は、依頼した浮動小数点コプロセッサ21がステージCWで格納する。以上によって、CPU1はステージDで命令をデコードしてコプロセッサ命令と判明し、かつコプロセッサ割込ステータス7がONと判明したので、ステージEの命令実行ステージで当該コプロセッサ命令のみを浮動小数点コプロセッサ21に通知し、実行を依頼する。これにより、CPU1はコプロセッサ命令と従来一緒に渡していた当該コプロセッサ命令のアドレスを渡す必要がなくなり、この分だけ処理が少なくなり、CPU1の性能を向上させることが可能となる。

【0034】次に、ステージCE:ステージDでCPU1からコプロセッサ命令C I ②を渡された浮動小数点コプロセッサ21は、このコプロセッサ命令C I ②を実行する。

【0035】ステージCW:実行した結果をメモリ3の所定のアドレスに格納する。この実行時に例外(オーバーフロー、アンダーフロー)が発生した場合、浮動小数点コプロセッサ21はコプロセッサ割込禁止ステータス7がONとここでは判明したので、最大値あるいは零を結果としてメモリ3の所定のアドレスに格納する。コプロセッサ例外信号はCPU6に通知しない。

【0036】以上によって、浮動小数点コプロセッサ21が依頼を受けたコプロセッサ命令の実行時に例外の発生に対応してコプロセッサ例外信号をCPU6に通知する必要がなく、CPU1に割込が発生しなく、処理が簡略化されることとなる。特に上述したように、コプロセッサ例外信号によりCPU1が割込処理でいずれのコプロセッサ命令で例外が発生したかを判別するためのアドレスが不要となり、依頼時にCPU1がコプロセッサ命令のアドレスを浮動小数点コプロセッサ21に通知する必要がなく、CPU1の処理が少なくなり、性能の低下を防止できる。

【0037】図3の(b)は、コプロセッサ割込許可状態(コプロセッサ割込禁止ステータス7がOFFの場合)のタイムチャートである。ここで、斜線の部分が本

実施例のC I (コプロセッサ命令)である。

【0038】図3の(b)において、ステージF:CPU1がメモリ3からメモリバス4を通して斜線を引いたコプロセッサ命令C I ②をフェッチする(読み込む)。ステージD:読み込んだコプロセッサ命令C I ②をデコードし、ここではコプロセッサ命令と判明すると共に、レジスタ11のコプロセッサ割込ステータス7がOFFを読み取って認識する。

【0039】ステージE:命令を実行する。ここでは、ステージDでコプロセッサ命令と判明し、コプロセッサ割込ステータス7がOFFと判明したので、このコプロセッサ命令C I ②および当該コプロセッサ命令C I ②のアドレスの両者を、コプロセッサバス5を通して浮動小数点コプロセッサ21に通知する。

【0040】ステージW:一般命令では結果を格納するが、ここではしない。結果は、依頼した浮動小数点コプロセッサ21がステージCWで格納する。以上によって、CPU1はステージDで命令をデコードしてコプロセッサ命令と判明したとき、コプロセッサ割込ステータス7がOFFと判明したので、ステージEの命令実行ステージで当該コプロセッサ命令およびアドレスの両者を浮動小数点コプロセッサ21に通知し、実行を依頼する。

【0041】次に、ステージCE:ステージDでCPU1からコプロセッサ命令C I ②を渡された浮動小数点コプロセッサ21は、このコプロセッサ命令C I ②を実行する。

【0042】ステージCW:実行した結果をメモリ3の所定のアドレスに格納する。この実行時に例外(オーバーフロー、アンダーフロー)が発生した場合、浮動小数点コプロセッサ21はコプロセッサ割込禁止ステータス7がOFFとここでは判明したので、コプロセッサ例外信号6をCPU1に通知する。そして、CPU1に割り込みが発生し、割り込み処理の中で浮動小数点コプロセッサ21が持つコプロセッサ命令およびそのアドレスをもとに所定の例外処理を行う。

【0043】

【発明の効果】以上説明したように、本発明によれば、CPU1が命令を順次取り込んでデコードしてコプロセッサ命令と判明したときにコプロセッサ割り込み禁止ステータス7がONのときにコプロセッサ命令のみをコプロセッサ2に送出し、当該コプロセッサ命令のアドレスの送出を抑止し、コプロセッサ2が依頼を受けたコプロセッサ命令を実行して例外が発生したときにコプロセッサ例外信号の送出を抑止してCPU1に通知しない構成を採用しているため、CPU1とコプロセッサ2の間のバス幅を増加させることなしに、コプロセッサ割込禁止ステータス7を設けてこれを操作して無用な命令アドレスの送出を無くし、CPU1の性能低下を防止することができる。

【図面の簡単な説明】

【図 1】 本発明の原理構成図である。

【図 2】 本発明の 1 実施例構成図である。

【図 3】 本発明のタイムチャート例である。

【図 4】 従来技術の説明図である。

【符号の説明】

1 : CPU

11 : レジスタ

2 : コプロセッサ

21 : 浮動小数点コプロセッサ

3 : メモリ

4 : メモリバス

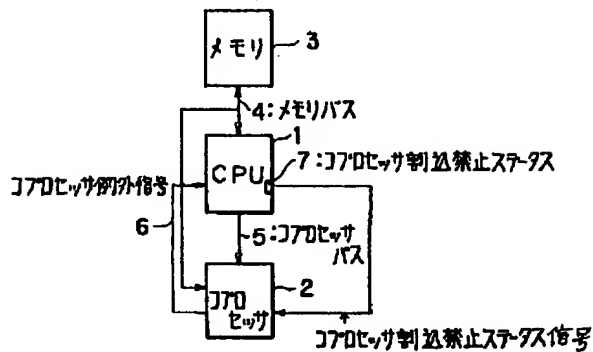
5 : コプロセッサバス

6 : コプロセッサ例外信号

7 : コプロセッサ割込禁止ステータス

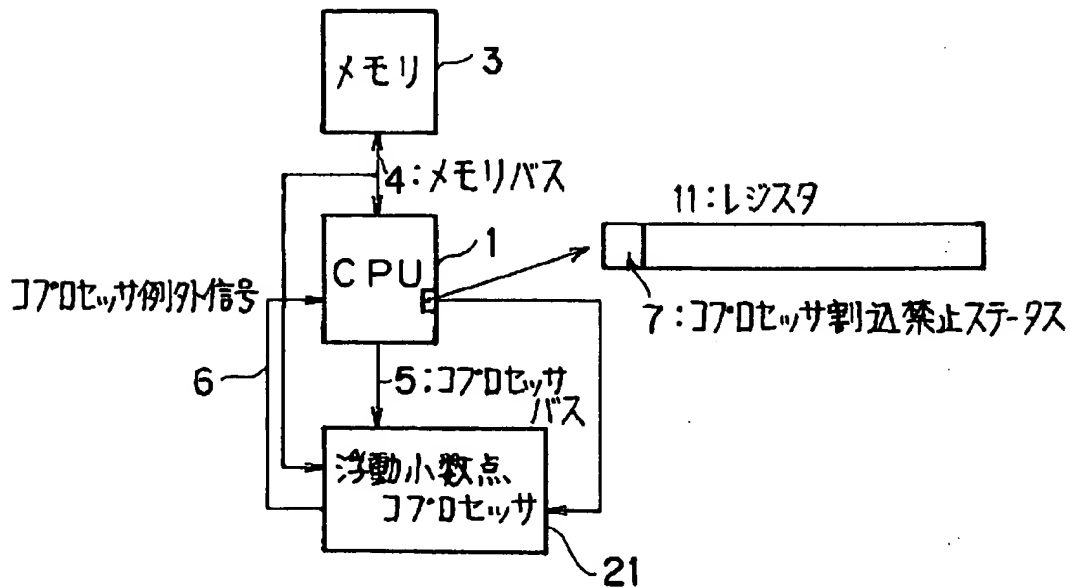
【図 1】

本発明の原理構成図



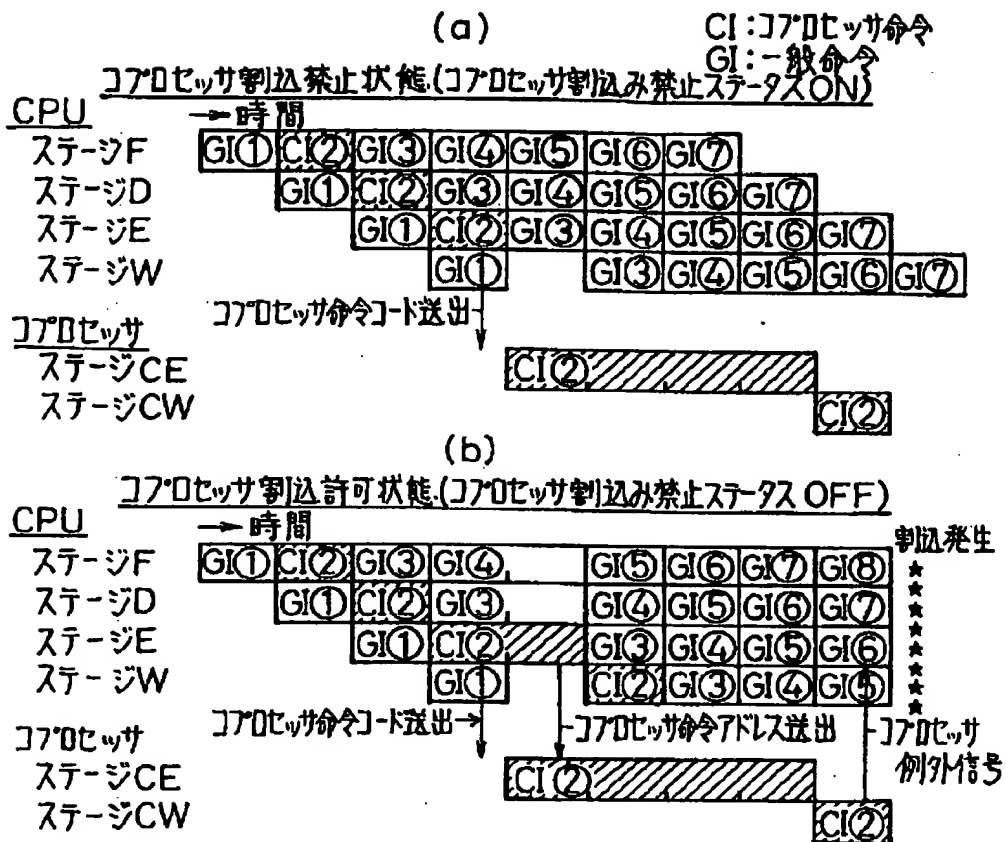
【図 2】

本発明の 1 実施例構成図



[図 3]

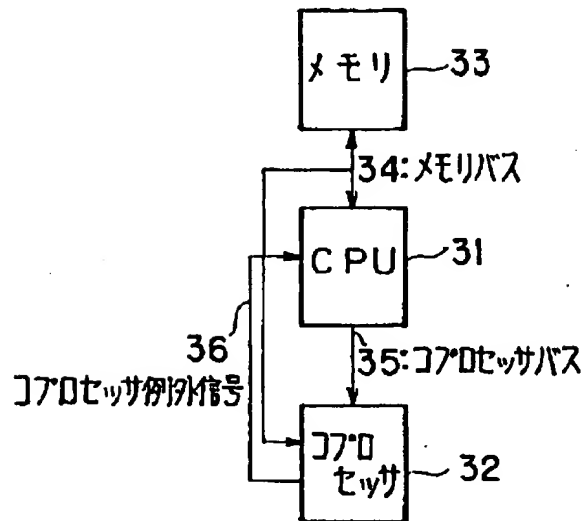
本 発 明 の タイムチャート例



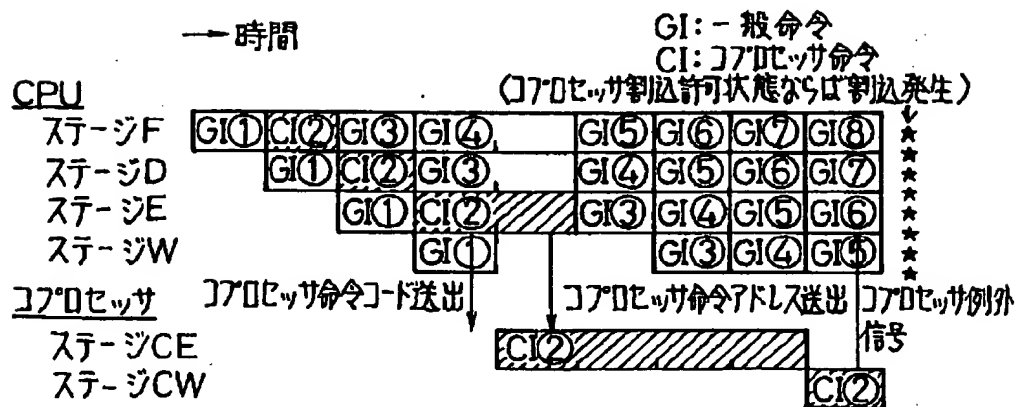
〔図 4〕

従来技術の説明図

(a)構成図



(b)タイムチャート



フロントページの続き

(72)発明者 野々村 一泰
神奈川県川崎市中原区上小田中1015
番地 富士通株式会社内

(72)発明者 渡部 徹
神奈川県川崎市中原区上小田中1015
番地 富士通株式会社内

(72)発明者 竹野 巧
神奈川県川崎市中原区上小田中1015
番地 富士通株式会社内

(72)発明者 加藤 慎哉
神奈川県川崎市中原区上小田中1015
番地 富士通株式会社内

(72)発明者 ボーンシャイ・チョンスワンナバイサー
ン
神奈川県川崎市中原区上小田中1015
番地 富士通株式会社内

(56)参考文献 特開 平2-306360 (JP, A)

(56)参考文献 特開 平2-306360(JP, A)

(58)調査した分野(Int.Cl.⁶, DB名)

G06F 9/38